

Fig. 1

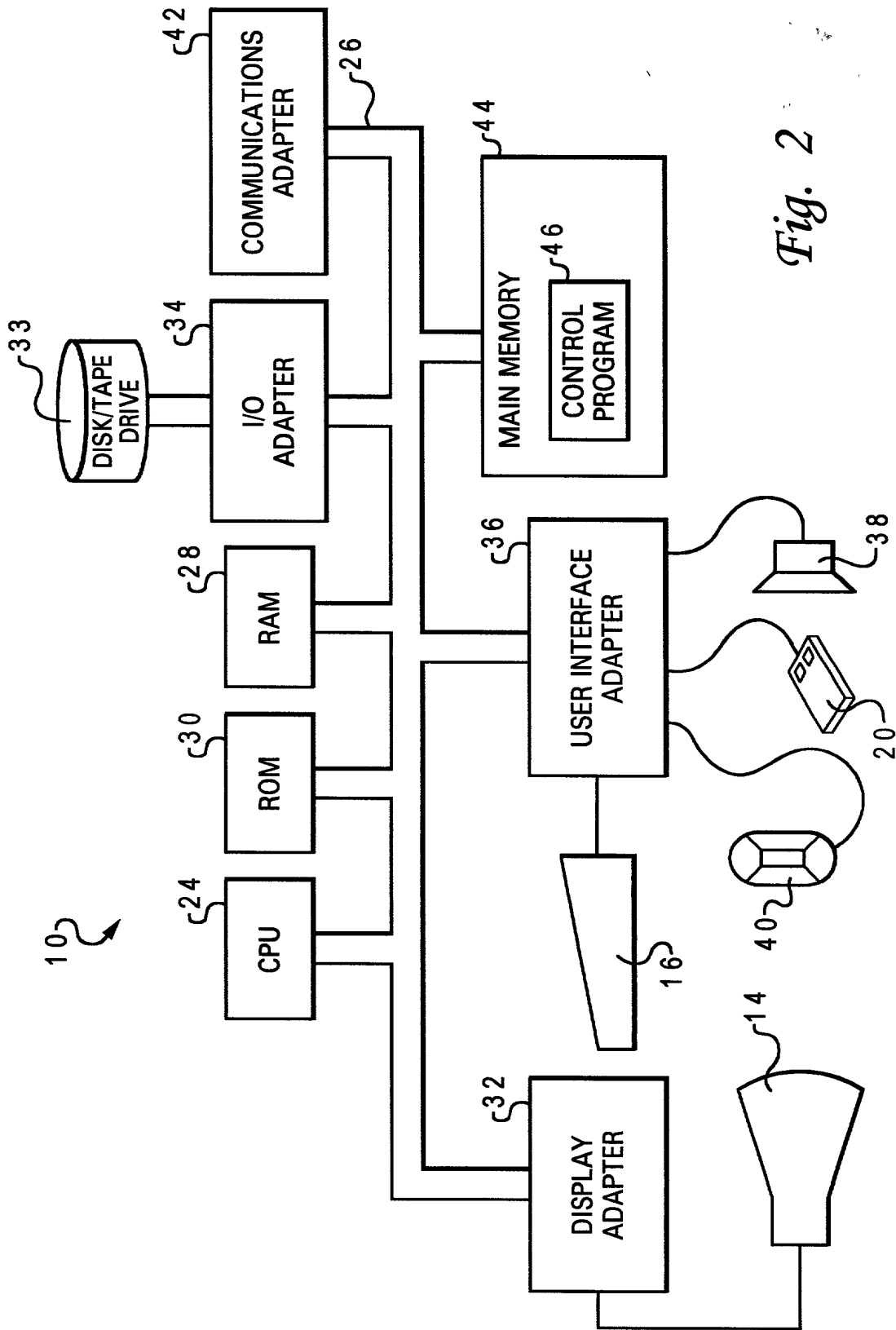


Fig. 2

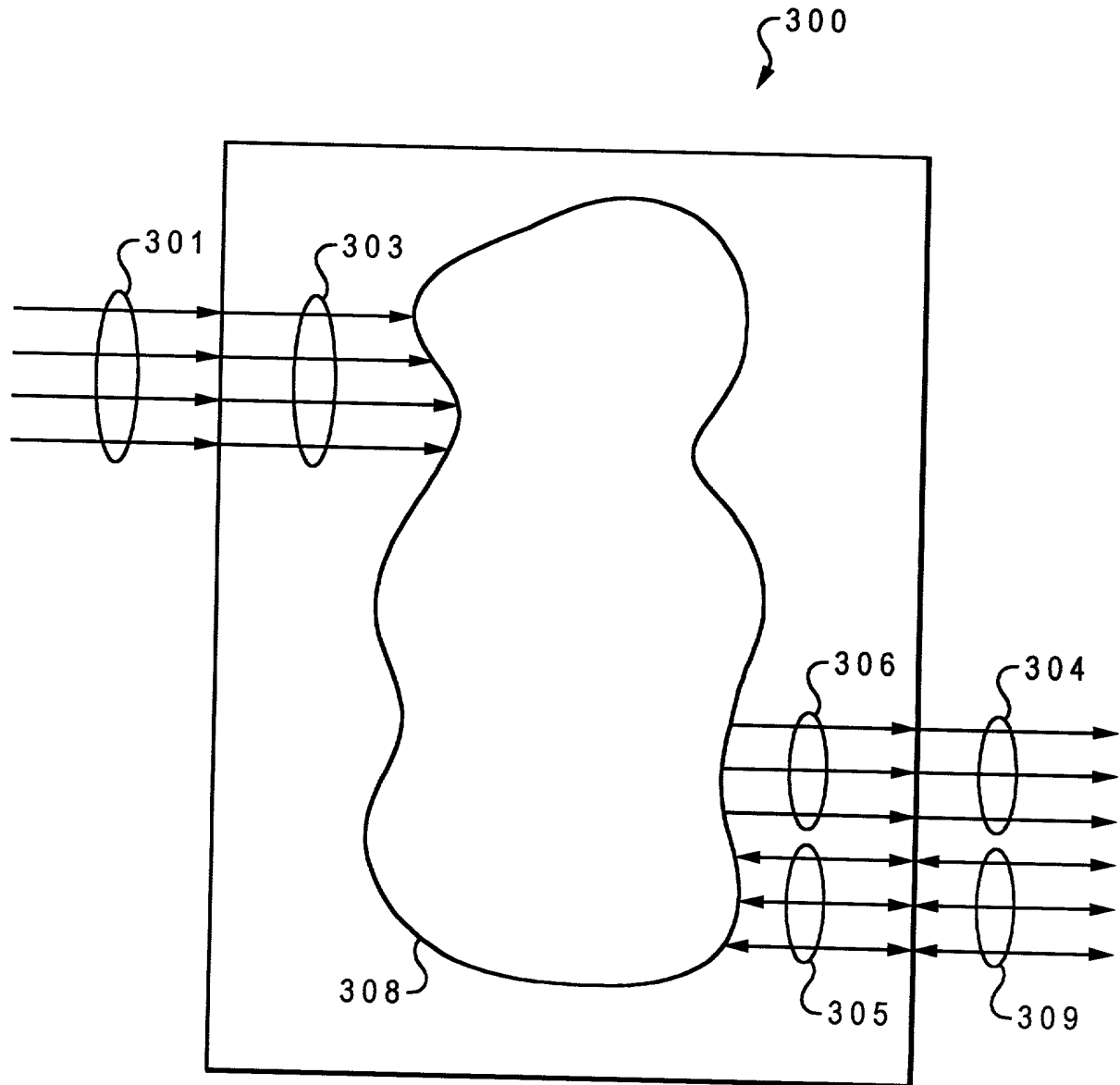


Fig. 3A

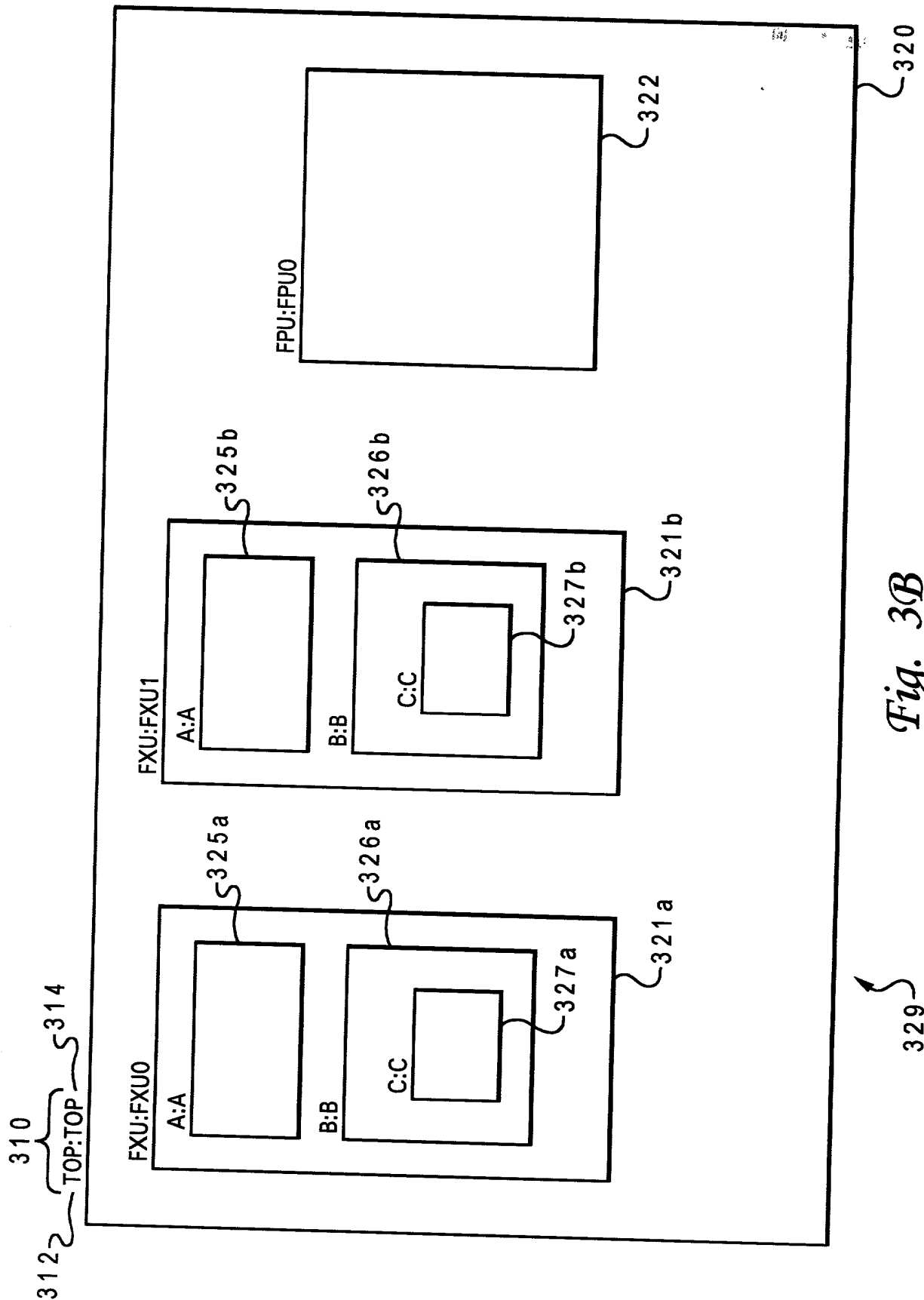
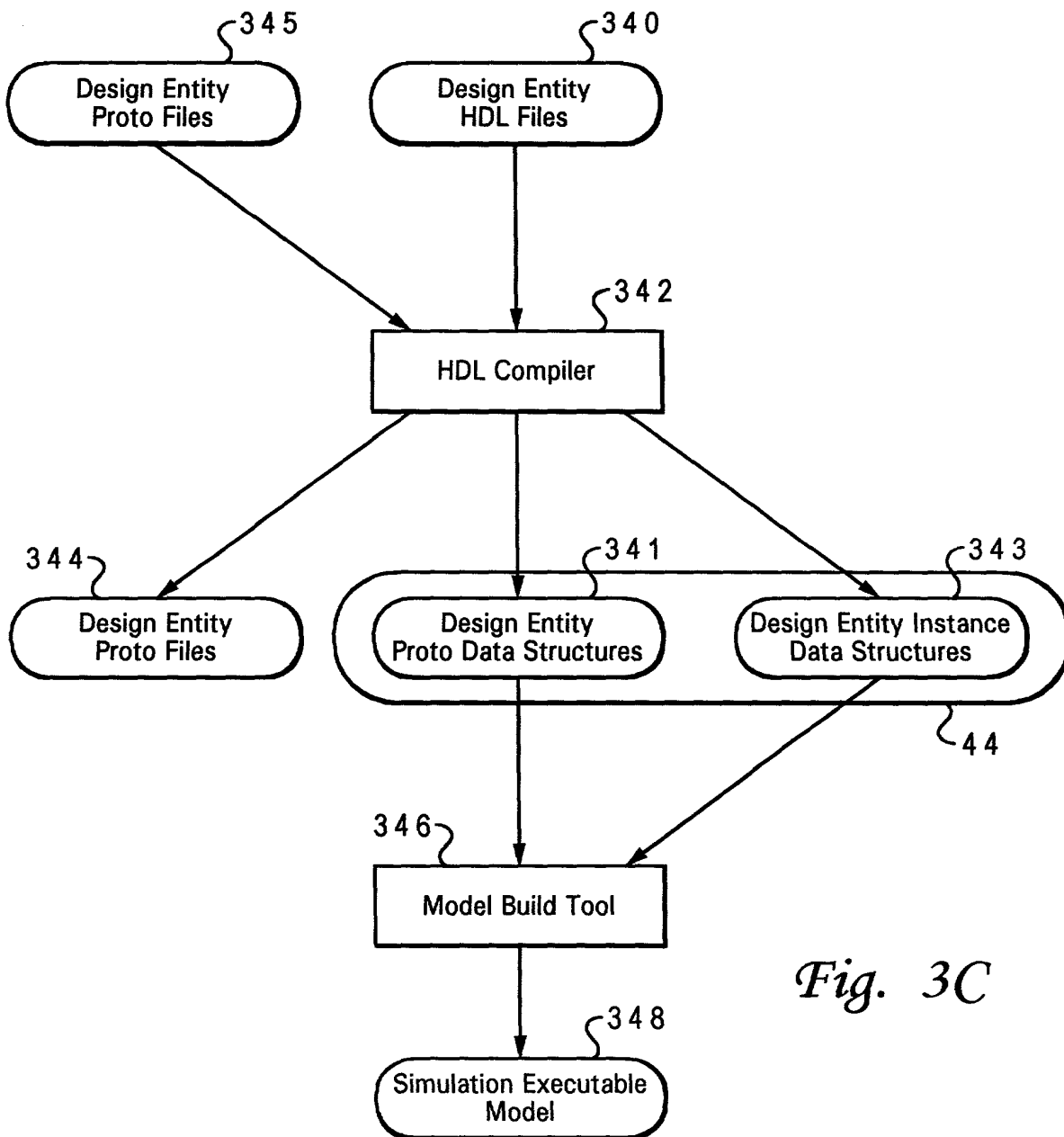


Fig. 3B

5/24

*Fig. 3C*

6/24

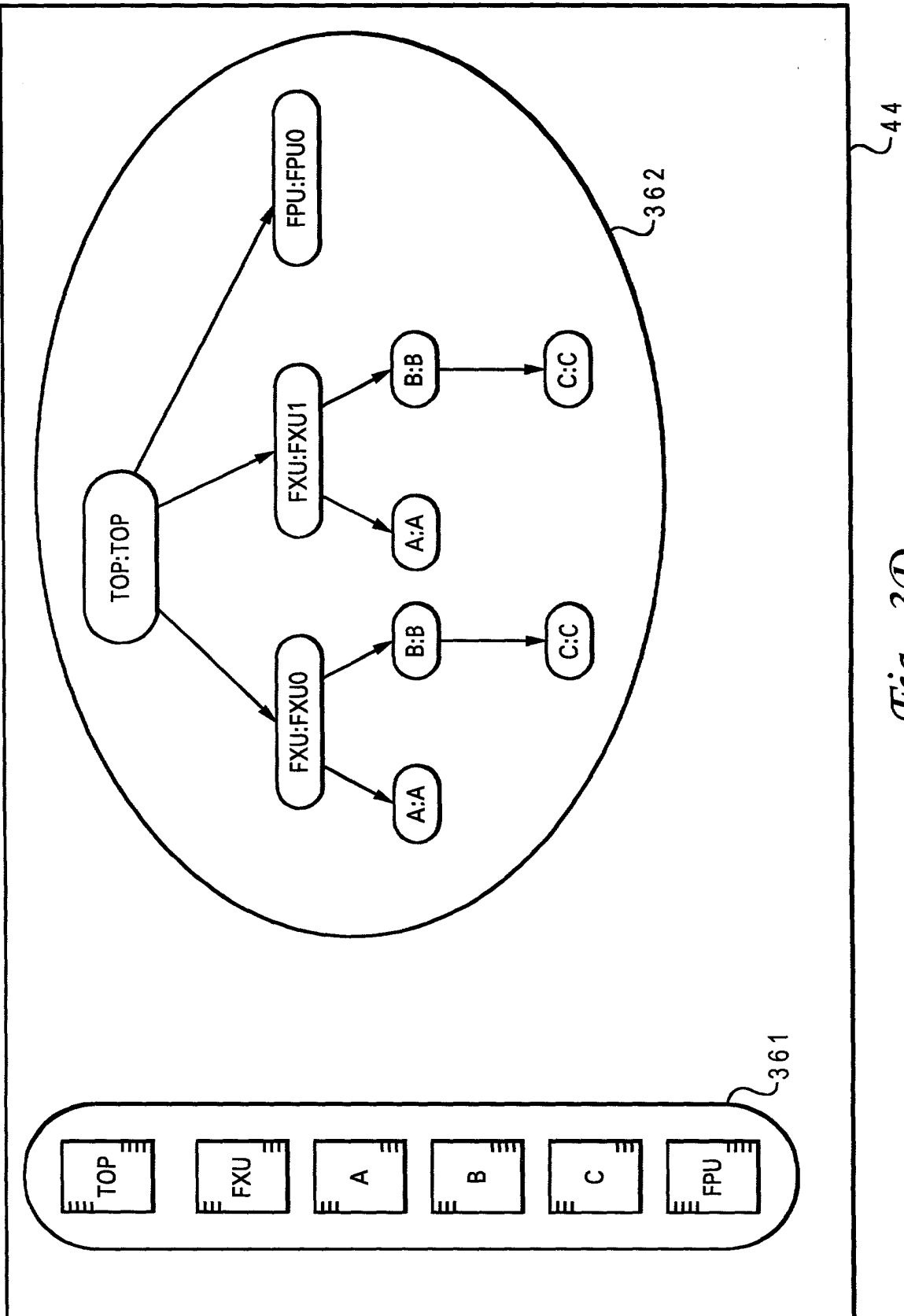
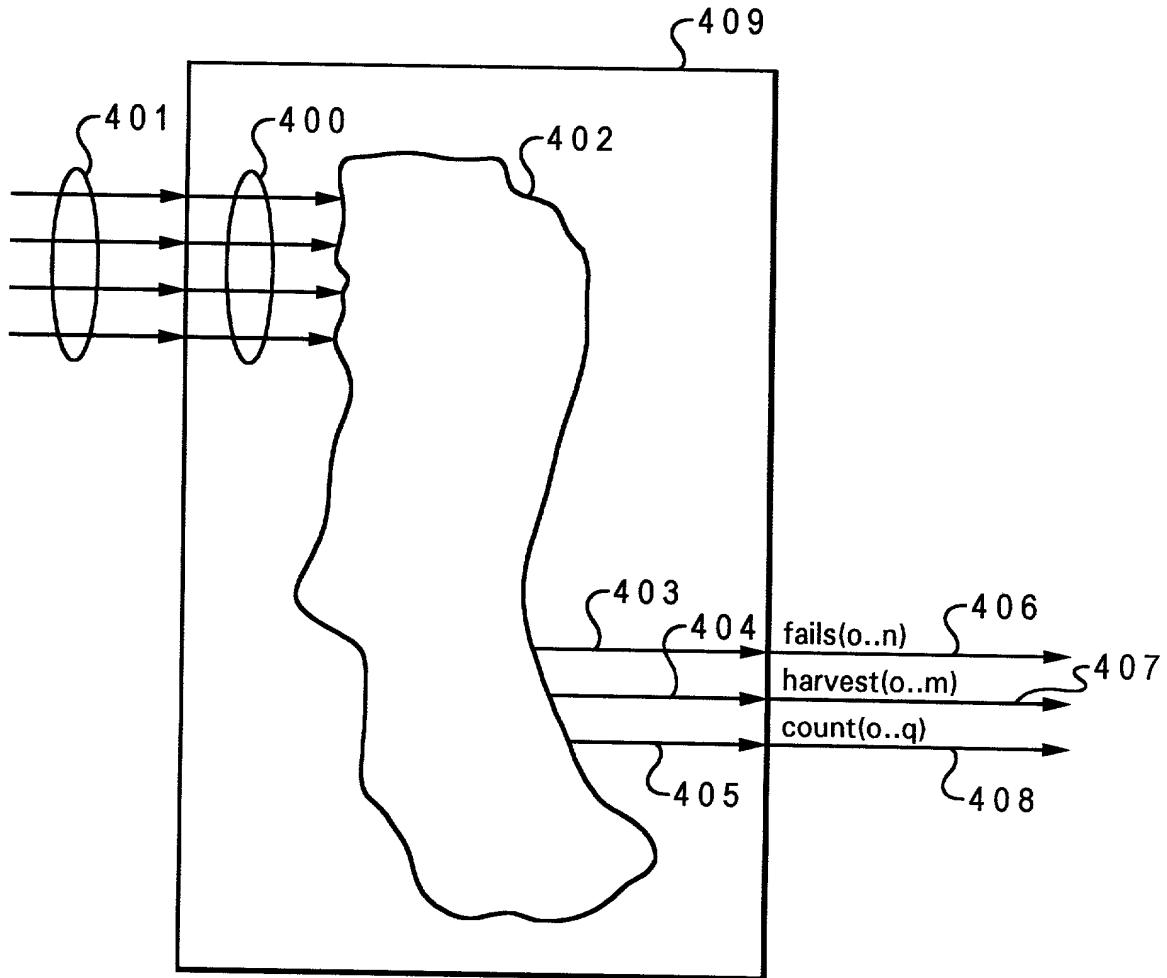


Fig. 3D

TOP:TOP:19999999

7/24

*Fig. 4A*

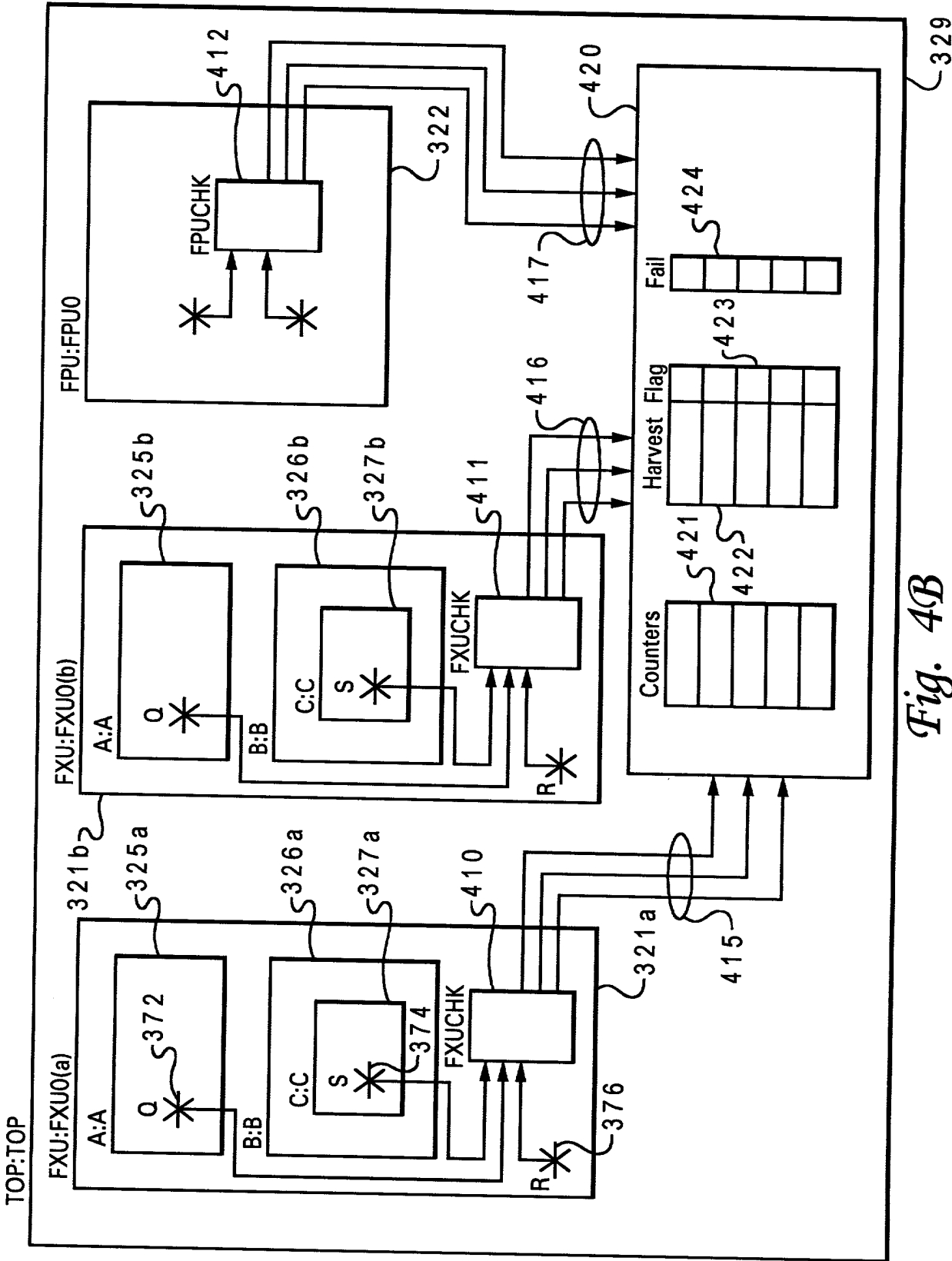


Fig. 4B

9/24

ENTITY FXUCHK IS

```

PORT(  S_IN      :    IN std_ulogic;
        Q_IN      :    IN std_ulogic;
        R_IN      :    IN std_ulogic;
        clock     :    IN std_ulogic;
        fails     :    OUT std_ulogic_vector(0 to 1);
        counts    :    OUT std_ulogic_vector(0 to 2);
        harvests  :    OUT std_ulogic_vector(0 to 1);
);

```

4 5 0

```

4 5 2 { --!! BEGIN
      --!! Design Entity: FXU;

```

```

4 5 3 { --!! Inputs
      --!! S_IN      =>    B.C.S;
      --!! Q_IN      =>    A.Q;
      --!! R_IN      =>    R;
      --!! CLOCK     =>    clock;
      --!! End Inputs

```

```

4 5 4 { --!! Fail Outputs;
      --!! 0 : "Fail message for failure event 0";
      --!! 1 : "Fail message for failure event 1";
      --!! End Fail Outputs;

```

```

4 5 5 { --!! Count Outputs;
      --!! 0 : <event0> clock;
      --!! 1 : <event1> clock;
      --!! 2 : <event2> clock;
      --!! End Count Outputs;

```

```

4 5 6 { --!! Harvest Outputs;
      --!! 0 : "Message for harvest event 0";
      --!! 1 : "Message for harvest event 1";
      --!! End Harvest Outputs;

```

```

4 5 7 { --!! End;

```

4 5 1

4 4 0

ARCHITECTURE example of FXUCHK IS

BEGIN

... HDL code for entity body section ...

END;

4 5 8

Fig. 4C

FOUO-4925460

10/24

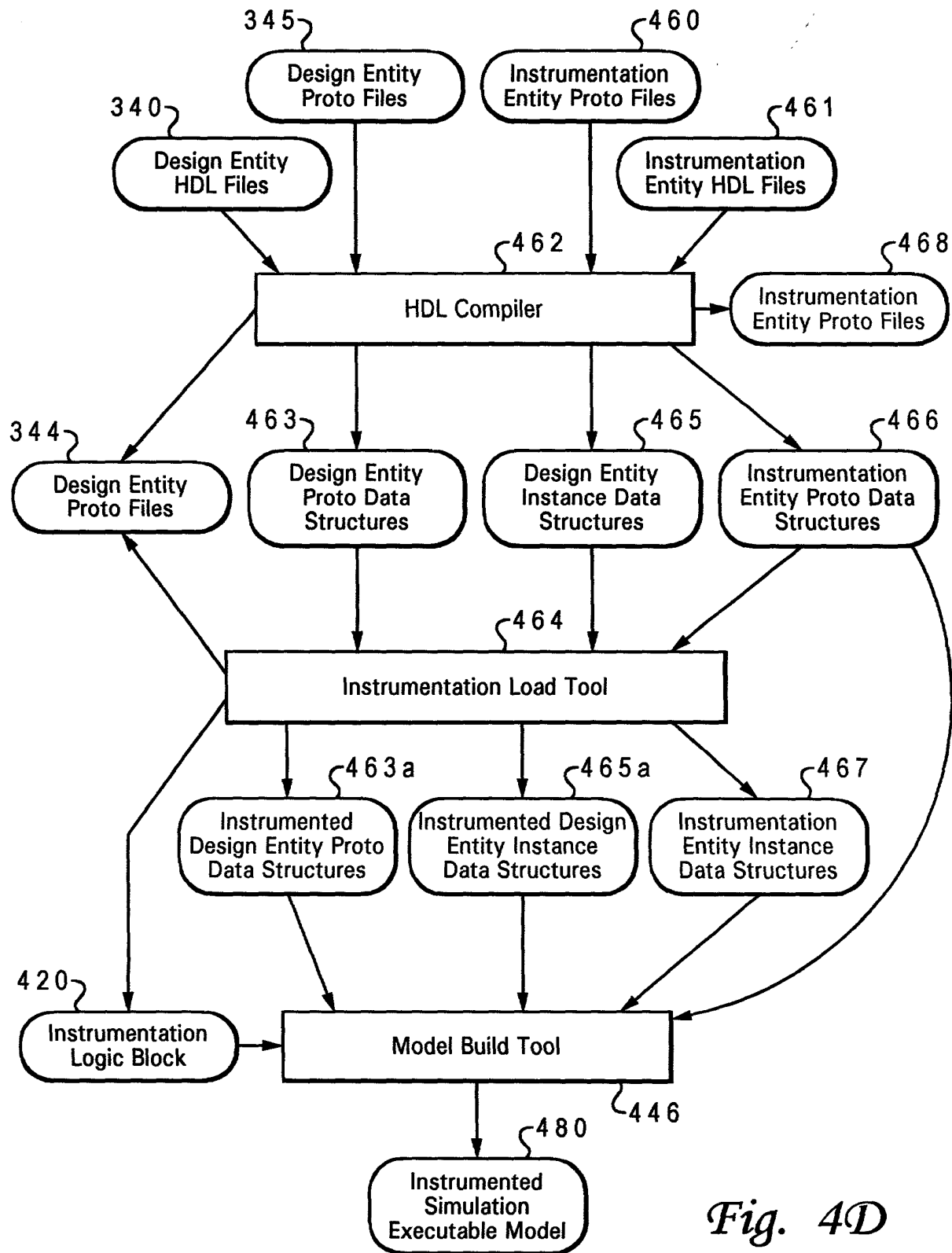


Fig. 4D

11/24

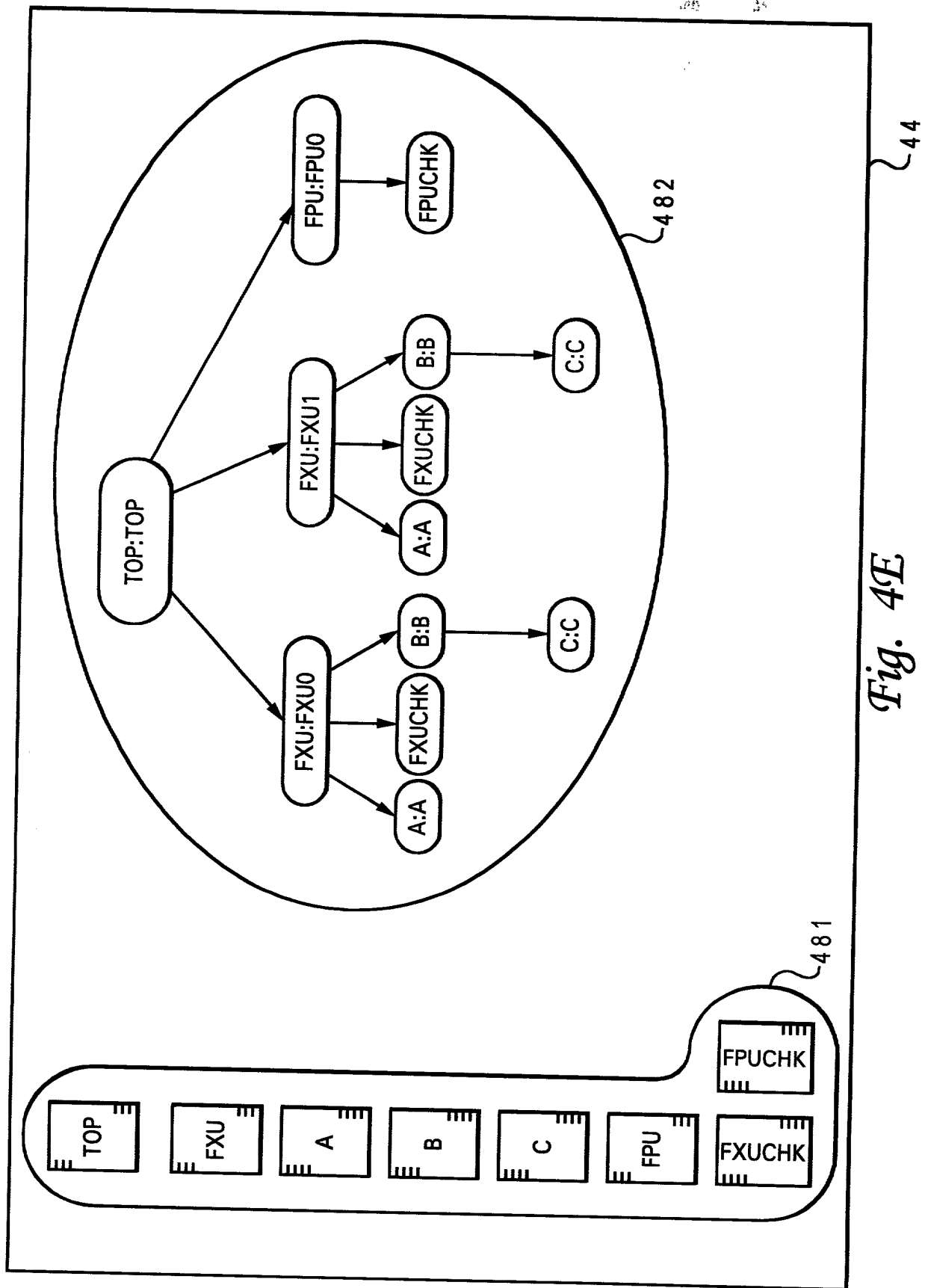


Fig. 4E

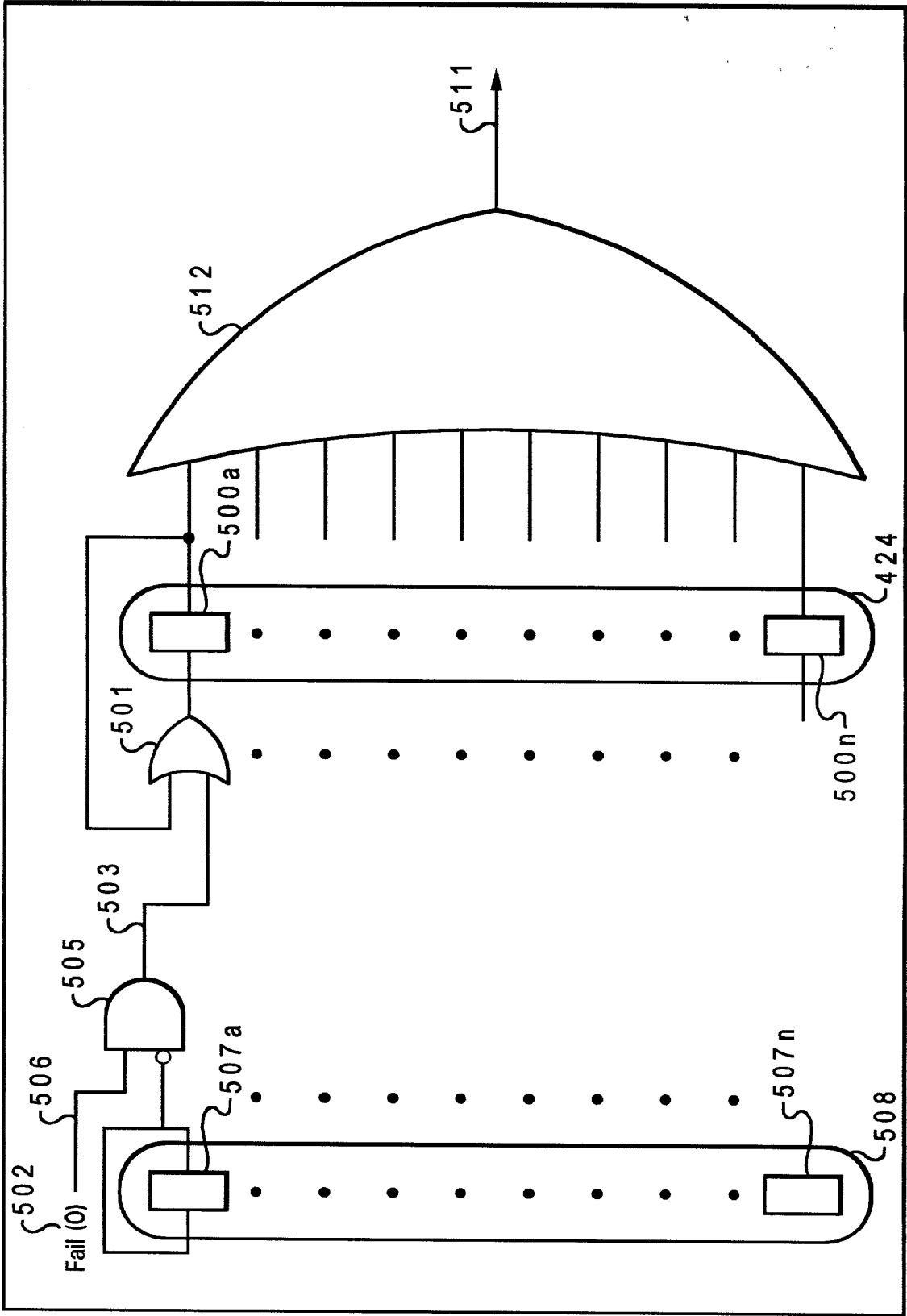


Fig. 5A

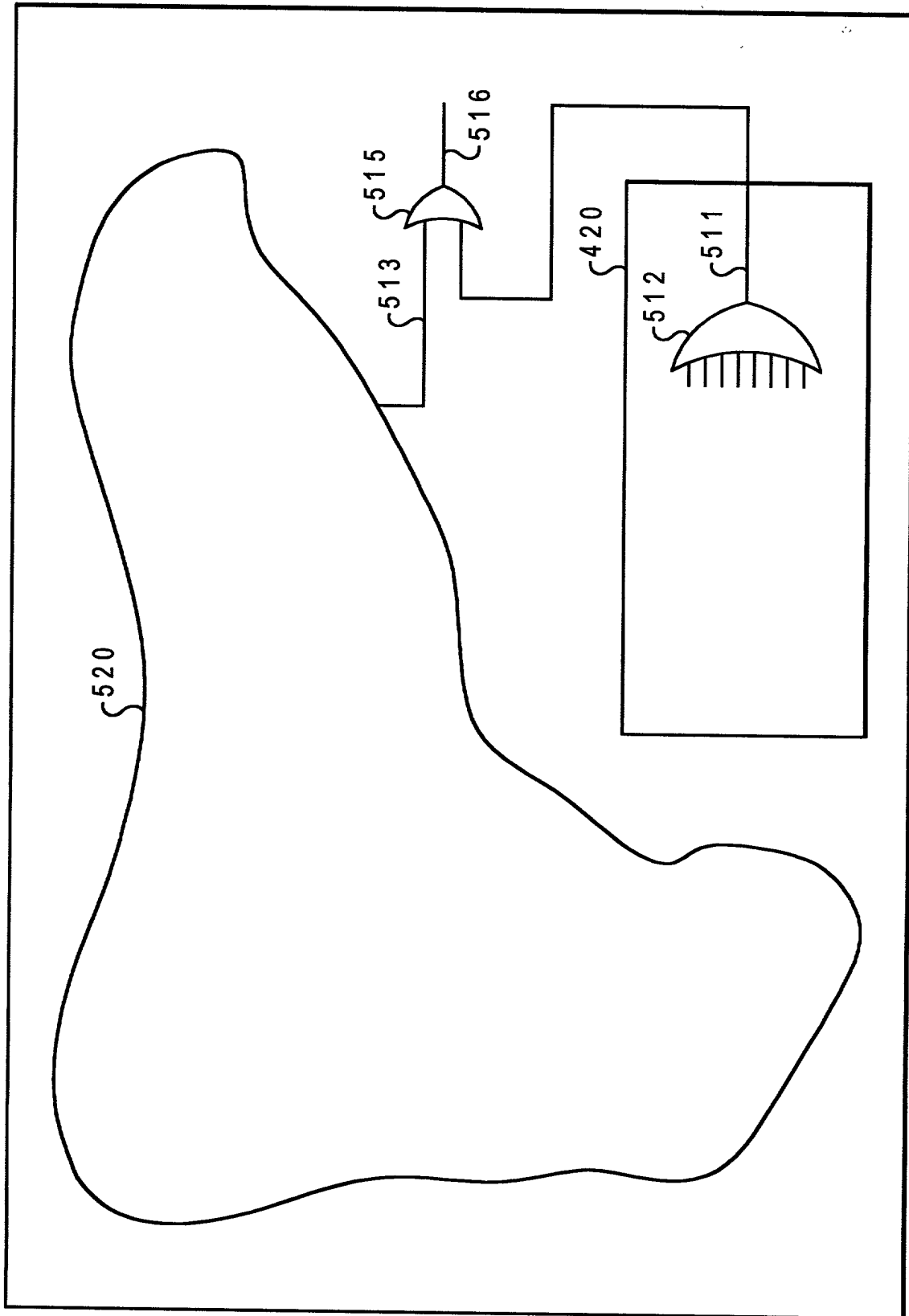


Fig. 5B

FIG. 5B

FIG. 6A

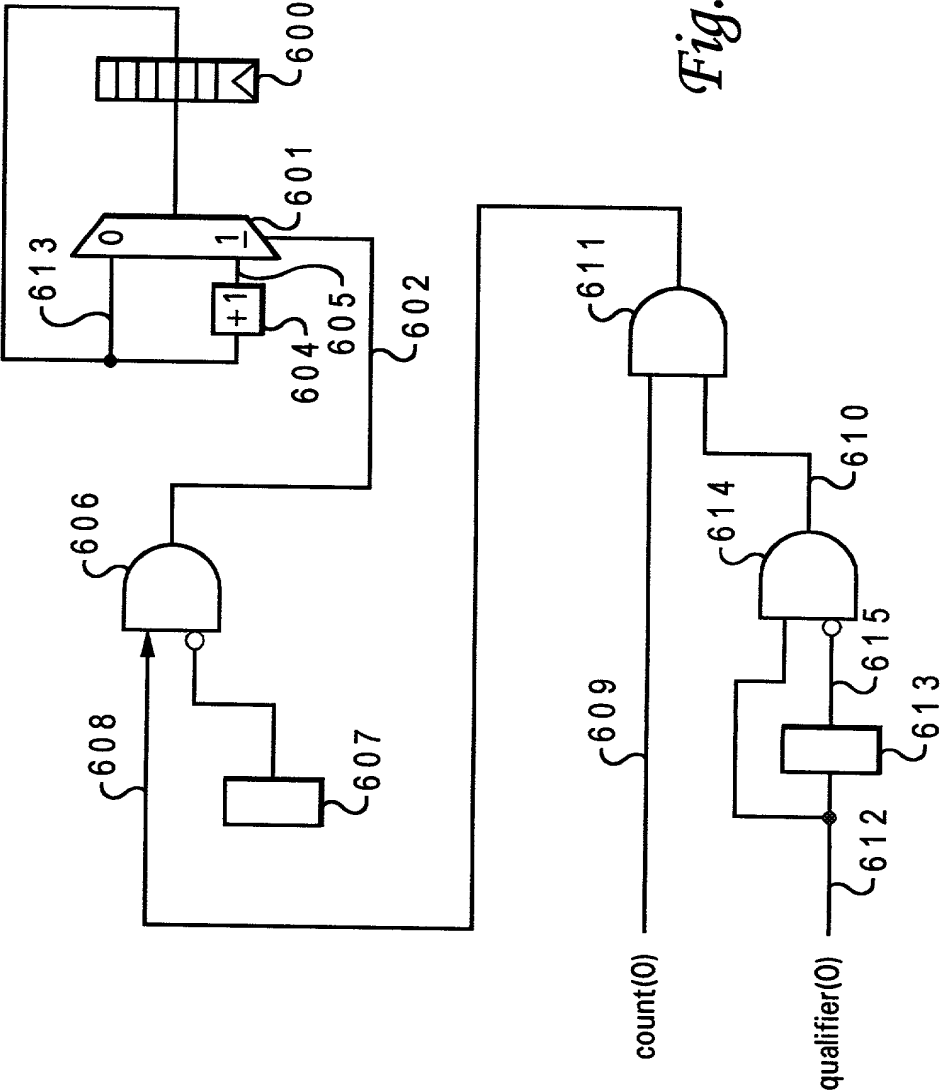
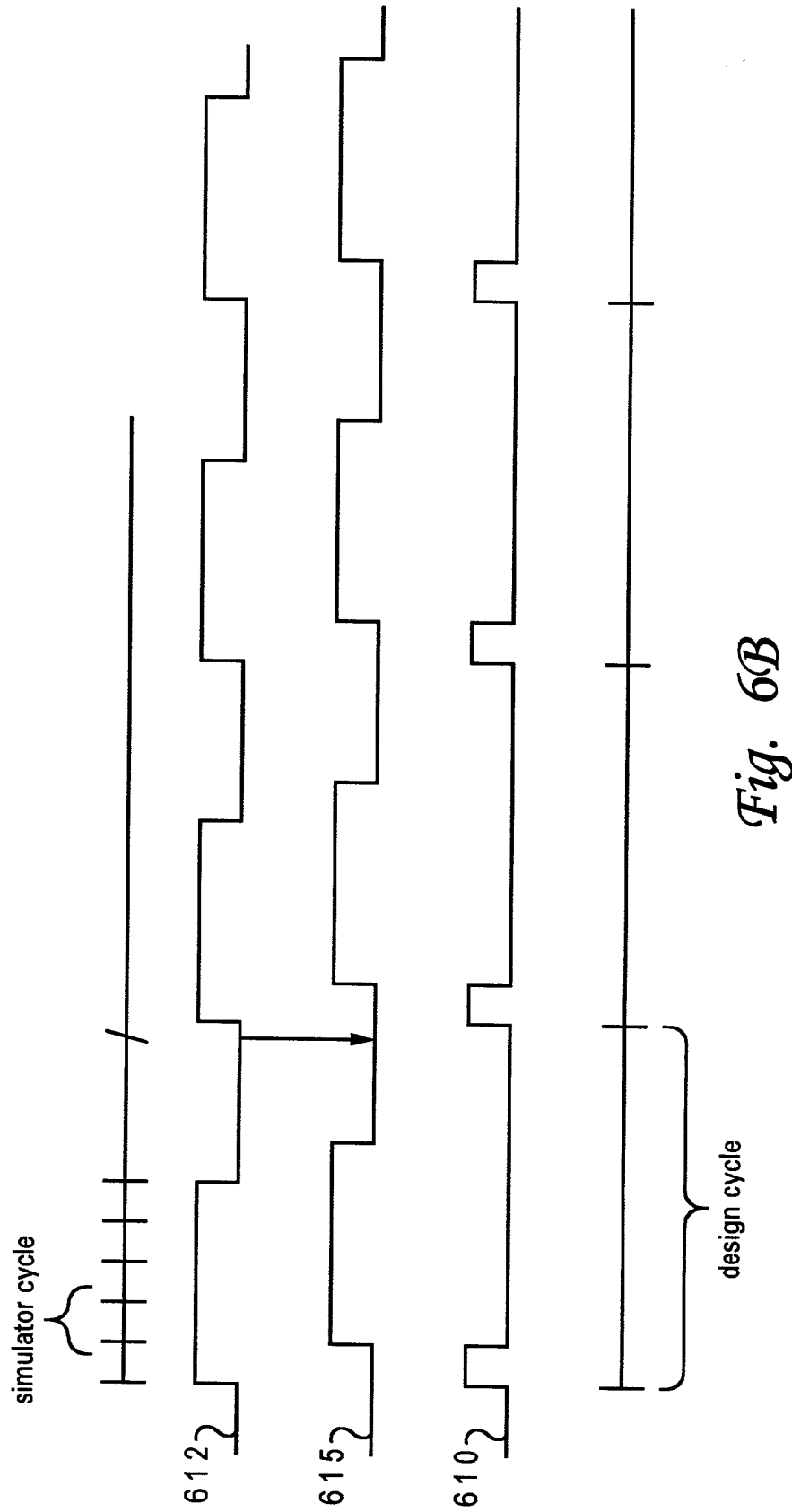


Fig. 6A

15/24



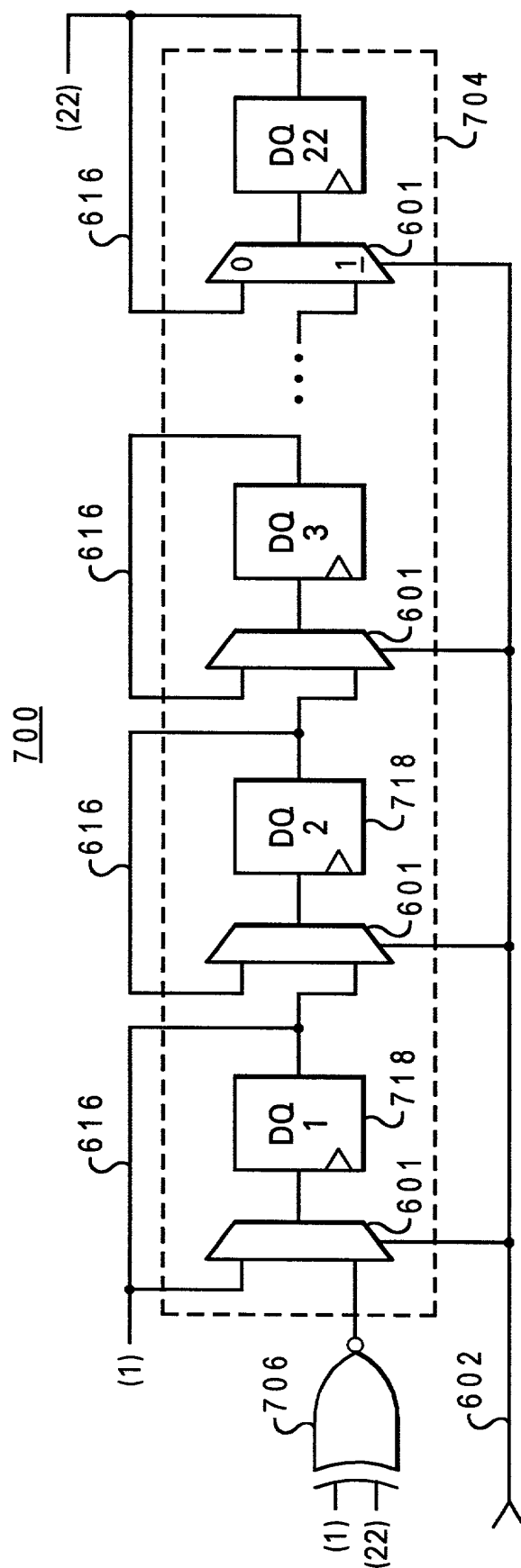
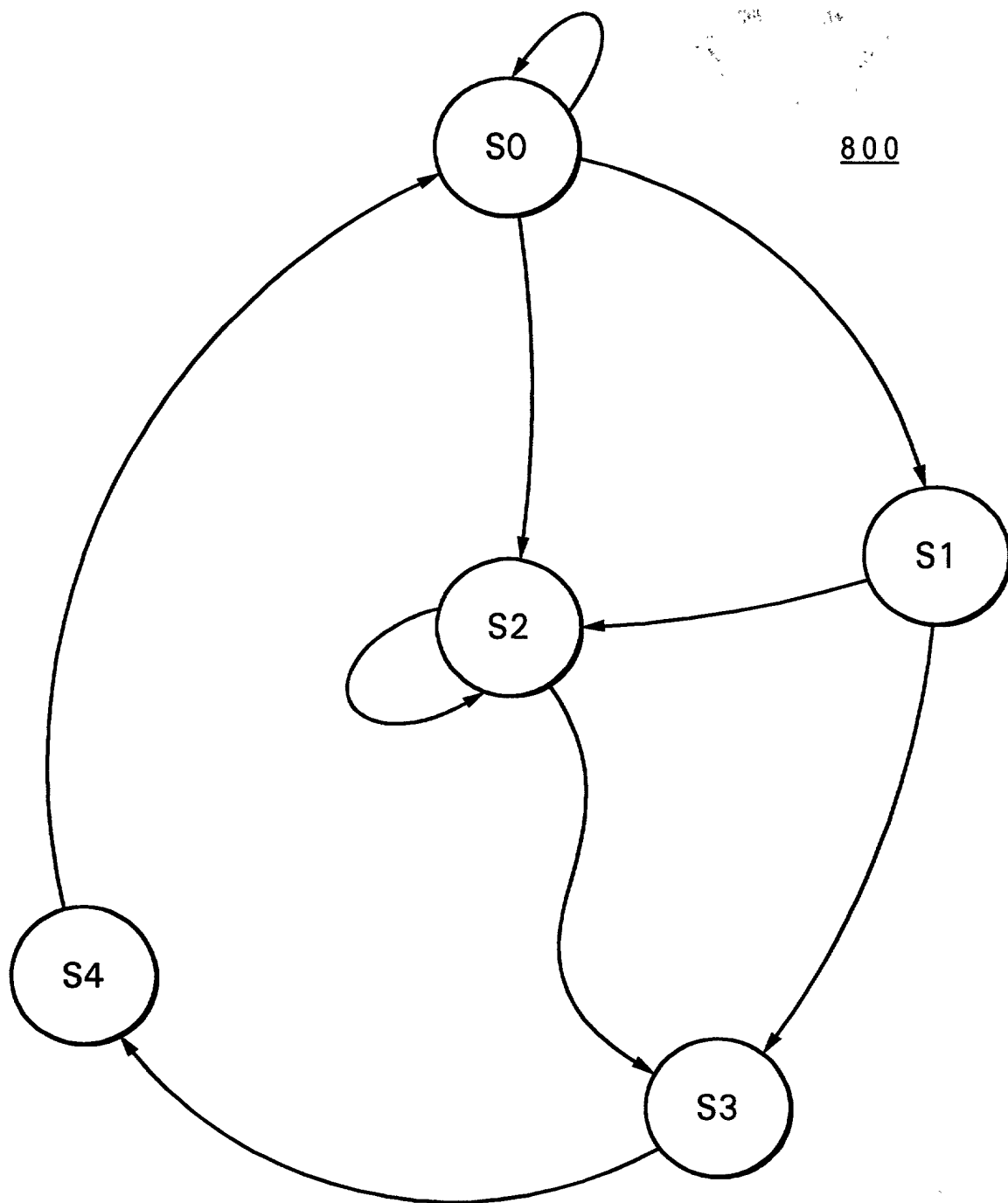


Fig. 7

17/24



800

Fig. 8A
Prior Art

FILED 11-03-2011

entity FSM : FSM

850

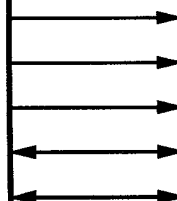
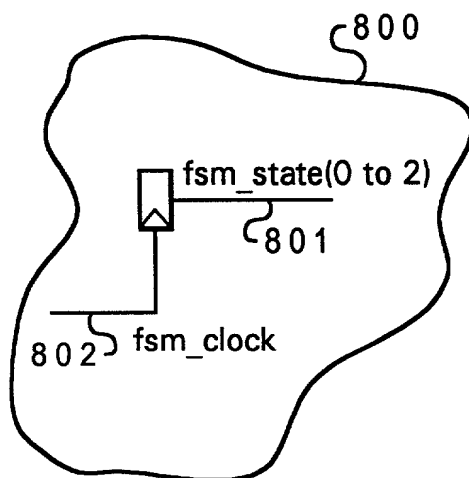
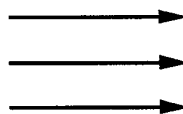


Fig. 8B
Prior Art

095384-034304
F06720-4925260

ENTITY FSM IS

PORT(
 ports for entity fsm....
);

ARCHITECTURE FSM OF FSM IS

BEGIN

 ... HDL code for FSM and rest of the entity ...

 fsm_state(0 to 2) <= ... Signal 801 ...

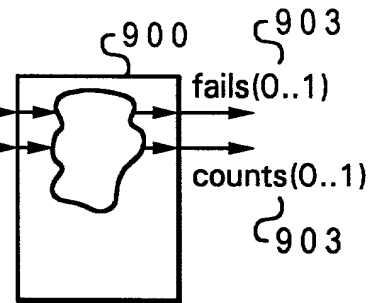
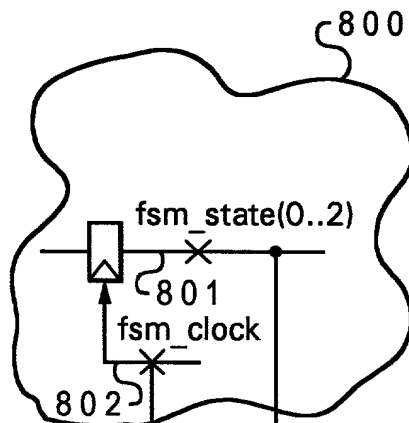
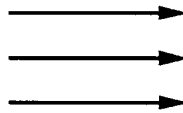
8 5 3 {	--!! Embedded FSM : examplefsm;	}	8 5 2	}	8 6 0
8 5 9 {	--!! clock : (fsm_clock);				
8 5 4 {	--!! state_vector : (fsm_state(0 to 2));				
8 5 5 {	--!! states : (S0, S1, S2, S3, S4);				
8 5 6 {	--!! state_encoding : ('000', '001', '010', '011', '100');				
8 5 7 {	--!! arcs : (S0 => S0, S0 => S1, S0 => S2,				
	--!! (S1 => S2, S1 => S3, S2 => S2,				
	--!! (S2 => S3, S3 => S4, S4 => S0);				
8 5 8 {	--!! End FSM;				

END;

Fig. 8C

entity FSM : FSM

850



\$examplefsm : \$examplefsm

902

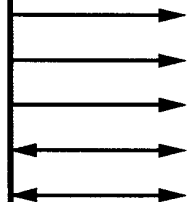
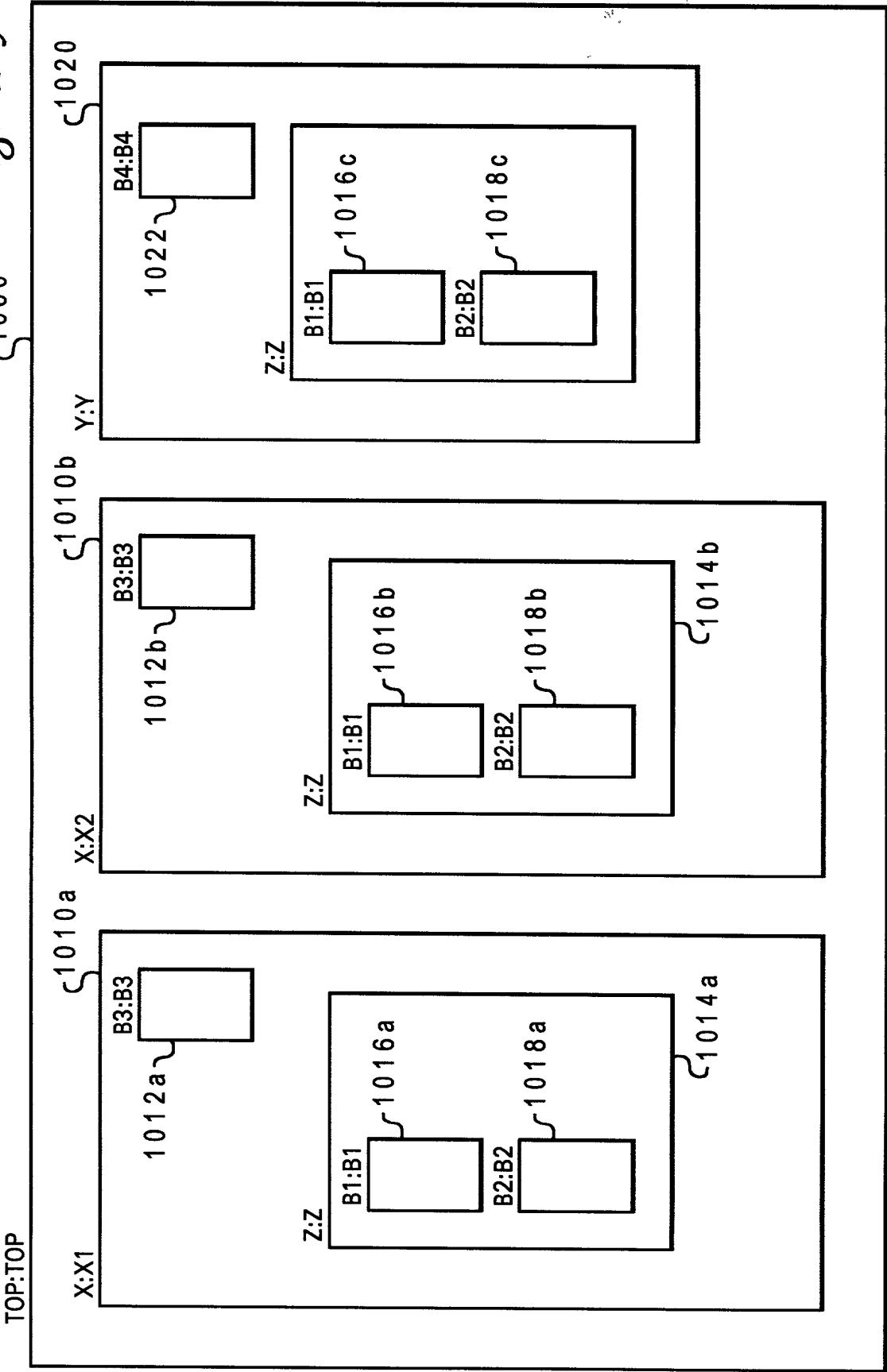


Fig. 9

Fig. 10A



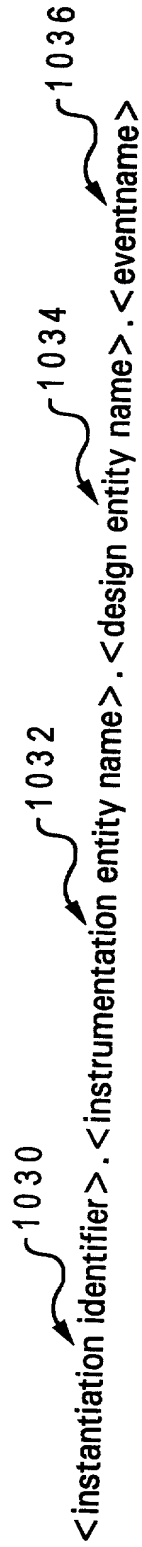


Fig. 10B

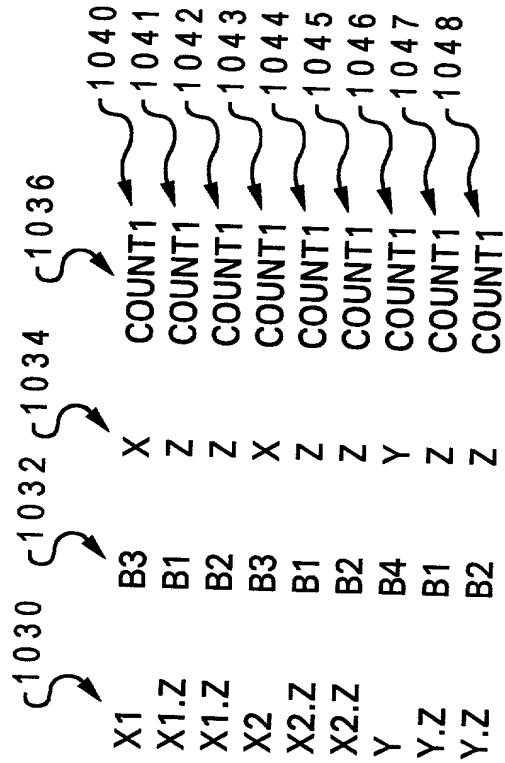
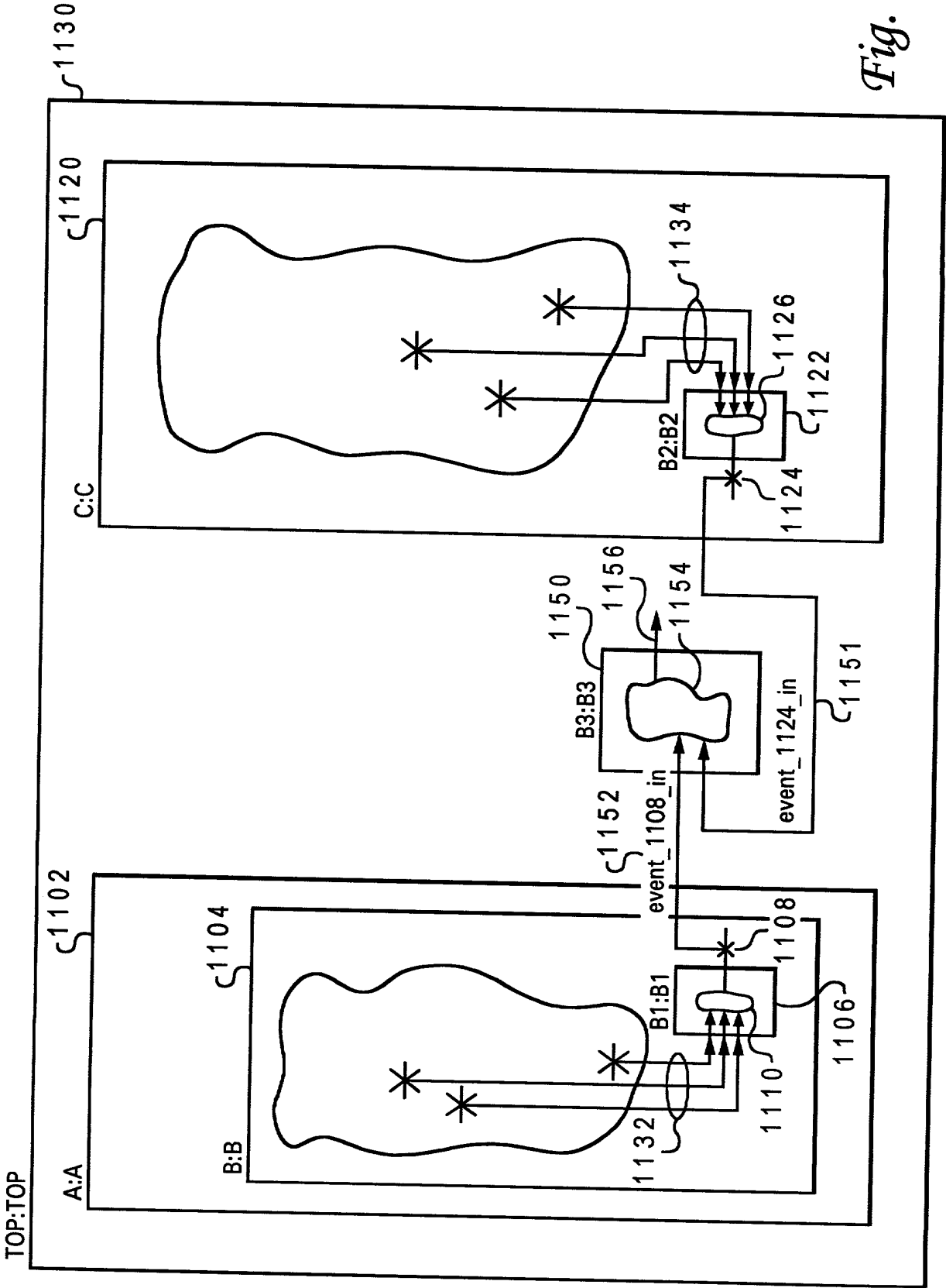


Fig. 10C



Fig. 10D

Fig. 11A



--!! Inputs
--!! event_1108_in <= C.[B2.count.event_1108];
--!! event_1124_in <= A.B.[B1.count.event_1124];
--!! End Inputs

1163 { 1165 } 1161
1164 { 1166 } 1162

Fig. 11B

--!! Inputs
--!! event_1108_in <= C.[count.event_1108];
--!! event_1124_in <= B.[count.event_1124];
--!! End Inputs

1171
1172

Fig. 11C